

# **CLAIM AMENDMENTS**

## **Claim Amendment Summary**

### **Claims pending**

- Before this Amendment: Claims 14-33.
- After this Amendment: Claims 14-22, 24-30, 32-33 and 50-52.

**Non-Elected, Canceled, or Withdrawn claims:** 1, 3-10, 12-13, 23, 31 and 34-49.

**Amended claims:** Claims 14, 19, 20, 28 and 30.

**New claims:** Claim 50-52.

---

**Claims:**

**1-13. (Canceled)**

**14. (Currently Amended)** A computer implemented method of managing access to a storage resource for one of a plurality of network-based applications in a multiple server storage system, the method comprising:

obtaining a resource identifier ~~associated with the storage resource~~ from a front end server;

utilizing ~~[[said]]~~ the resource identifier to ~~lookup~~ locate, in a resource lookup store of a lookup partitioning service server, a partition of a storage server ~~where said storage resource is located~~ associated with the resource identifier; ~~[[and]]~~

in an event said partition of said storage server is associated with the resource identifier, granting access to the storage resource by providing ~~[[said]]~~ a location of said partition of said storage server to said front end server; and

in an event said partition of said storage server is not associated with the resource identifier:

creating a new storage resource in a storage partition;

associating the resource identifier with said storage partition in said resource lookup store; and

providing a location of said storage partition to said front end server.

**15. (Previously Presented)** The method of Claim 14, wherein said location of said partition of said storage server is on one of a plurality of storage servers.

**16. (Previously Presented)** The method of Claim 14, including a plurality of storage partitions, said plurality including a primary storage partition and a redundant storage partition each containing said storage resource.

**17. (Previously Presented)** The method of Claim 16, wherein said primary storage partition and said redundant storage partition are each located on separate storage servers of said plurality of storage.

**18. (Previously Presented)** The method of Claim 17, wherein, if the primary storage partition is unavailable, the storage server location is the redundant storage partition.

**19. (Currently Amended)** The method of Claim 14, further comprising determining which lookup partitioning service server of a plurality of lookup partitioning service servers will provide said looked-up storage server location in response to [[said]] the resource identifier.

**20. (Currently Amended)** The method of Claim 19, wherein determining which lookup partitioning service server will provide said looked-up storage server

location comprises processing [[said]] the resource identifier ~~though~~ through a hash function to provide a hashed resource identifier associated with a particular lookup partitioning service server.

**21. (Previously Presented)** The method of Claim 20, wherein each lookup partitioning service server is associated with a predetermined set of hashed resource identifiers.

**22. (Previously Presented)** The method of Claim 14, further comprising moving the storage resource from one storage partition to a new storage partition and updating said resource lookup store with said new storage partition.

**23. (Canceled)**

**24. (Previously Presented)** The method of Claim 14, further comprising calculating a load balancing factor for each storage partition of a plurality of storage partitions and using said load balancing factors to determine the storage partition in which said new storage resource should be created.

**25. (Previously Presented)** The method of Claim 24, wherein said load balancing factor is based on value selected from the values consisting of: a mapping number, a count of mapping accesses, and a manual weighting value.

26. **(Previously Presented)** The method of Claim 24 further comprising adjusting a manual weighting value to increase the usage of said one of said storage servers.

27. **(Previously Presented)** The method of Claim 24, further comprising adjusting a manual weighting value to decrease the usage of said one of said storage servers.

28. **(Currently Amended)** A computer readable medium containing computer-executable instructions for performing the method of managing access to a storage resource for one of a plurality of network-based applications in a multiple server storage system, the computer-executable instructions comprising instructions for:

receiving a resource identifier associated with the storage resource from a front end server;

utilizing the resource identifier to lookup, in a resource lookup store of a lookup partitioning service server, a storage partition associated with the resource identifier;

in an event the storage partition is associated with the resource identifier:

locating ~~[[a]]~~ the storage partition of a storage server ~~where said storage resource is located utilizing said resource identifier in a lookup store of a lookup partitioning service server;~~ and

sending [[said]] the location of said storage partition of said storage server  
to said front end server to grant access to said storage resource; and  
in an event the storage partition is not associated with the resource identifier:  
failing to locate a mapping to the storage resource;  
creating a new storage resource in a new storage partition;  
mapping the resource identifier to said new storage partition in said  
resource lookup store; and  
sending a location of said new storage partition to said front end server.

29. **(Previously Presented)** The method of Claim 28, further comprising determining which lookup partitioning service server of a plurality of lookup partitioning service servers will locate said storage partition in response to said resource identifier.

30. **(Currently Amended)** The method of Claim 28, further comprising relocating the storage resource from one storage partition to ~~a new~~ a different storage partition and updating the mapping of [[said]] the resource identifier at said lookup partitioning service server.

31. **(Canceled)**

32. **(Previously Presented)** The method of Claim 28, further comprising calculating a load balancing factor for each storage partition of a plurality of storage

partitions and using said load balancing factors to determine the storage partition in which said new storage resource should be created.

**33. (Previously Presented)** The method of Claim 32, wherein said load balancing factor is based on value selected from the values consisting of: a mapping number, a count of mapping accesses, and a manual weighting value.

**34-49. (Canceled)**

**50. (New)** A lookup partitioning server comprising:

a processing unit;

at least one primary lookup partition; and

at least two redundant lookup partitions which mirror two respective different primary lookup partitions stored on other look-up partitioning servers;

a memory configured to store computer-executable instructions configured to manage access to a plurality of storage resources at a plurality of storage servers, the computer-executable instructions performing acts comprising:

receiving a resource identifier associated with a storage resource from a front end server;

utilizing the resource identifier to lookup, in a resource lookup store, a storage partition associated with the resource identifier;

in an event a particular storage partition is associated with the resource identifier:

locating the particular storage partition of a storage server; and  
sending a location of the particular storage partition of the storage server to the front end server to grant access to the storage resource; and

in an event no particular storage partition is associated with the resource identifier:

failing to locate a mapping to the storage resource;  
determining a load balancing factor for each storage partition of a plurality of storage partitions;  
using the load balancing factors to determine a new storage partition in which a new storage resource should be created and creating the new storage resource in the new storage partition;  
mapping the resource identifier to the new storage partition in the resource lookup store; and  
sending a location of the new storage partition to the front end server.

**51. (New)** The lookup partitioning server of claim 50, wherein determining a load balancing factor for each storage partition comprises:

given  $n$  storage partitions, determining the number of mapping counts  $C$  for each of the  $n$  storage partitions; and



calculating a load balancing factor ( $LBF_m$ ) for each storage partition  $m$  using the following:

$$LBF_m = (1/C_m) / (1/C_1 + 1/C_2 + \dots + 1/C_n).$$

**52. (New)** The lookup partitioning server of claim 51, wherein using the load balancing factors to determine a new storage partition comprises:

generating a random number  $R$ , where  $R$  is a real number between 0 and 1;

ranking the load balancing factors in ascending order;

locating a particular storage partition, where the sum of all the load balancing factors ranked lower than the load balancing factor corresponding to the particular storage partition is less than or equal to  $R$ , and the sum of all the load balancing factors ranked lower than the load balancing factor corresponding to the particular storage partition plus the load balancing factor corresponding to the particular storage partition is greater than  $R$ .